(51) International Patent Classification ^{6} :		(11) International Publication Number: WO 97/06477
G06F	A2	(43) International Publication Date: 20 February 1997 (20.02.97)

(54) Title: METHOD AND APPARATUS FOR GENERATING IMAGE TEXTURES

(57) Abstract

A method and apparatus for synthesizing texture images that match the texture appearance of a given digitized sample of an image. This synthesis is completely automatic and requires only the "target" texture image as an input. It allows generation of as much texture as desired so that any object or surface can be covered with the synthesized texture. The apparatus generates texture parameters for the target texture sample image that reflect the texture in that image. These texture parameters are used to modify the noise image such that the modified noise image attains the texture of the target texture image. To further improve the texture of the modified noise image, an image pyramid is constructed for both the modified noise image and the target texture sample image. The apparatus then iteratively updates the modified noise image on a coarse-to-fine basis using the subbands of the image pyramids. The output image is an updated modified noise image that has attained the texture of the target texture sample image.

METHOD AND APPARATUS FOR
GENERATING IMAGE TEXTURES

The invention relates to techniques for image processing. More particularly,
the invention relates to a method and apparatus for generating image textures.

BACKGROUND OF THE DISCLOSURE

Generally speaking, computer renderings of objects with surface texture are
more interesting and realistic than those without texture. Texture mapping is a
technique that adds an appearance of surface detail by wrapping or projecting a
digitized texture image onto a computer generated surface. Digitized textures can
be obtained from a variety of sources, e.g., cropped from a photo CD image as a
texture "chip", but the resulting texture chip may not have a desired size or shape
corresponding to a surface that is to be textured. Furthermore, to cover a large
object, an image rendering system may need to repeat the texture; this can lead to
unacceptable artifacts either in the form of visible seams, visible repetition, or
both.

Texture mapping suffers from an additional fundamental problem; often
there is no natural map from the (planar) texture image to the geometry/topology
of the surface, so the texture may be distorted unnaturally when mapped. There
are some partial solutions to this distortion problem but there is no universal
solution for mapping an image onto an arbitrarily shaped surface.

An alternative to texture mapping is to create (paint) textures by hand
directly onto a computer generated, three-dimensional surface model, but this
process is both labor intensive and requires considerable artistic skill.

Another alternative is to use computer-synthesized textures so that as
much texture can be generated as needed to cover a particular surface.
Furthermore, some of the synthesis techniques produce textures that tile
seamlessly.

Using synthetic textures, the distortion problem has been solved in two
different ways. First, some techniques work by synthesizing texture directly on
the object surface. The second solution is to use so-called "solid textures". A solid
texture is a three-dimensional array of color values. A point on the surface of an
object is colored by the value of the solid texture at the corresponding
three-dimensional point. Solid texturing can be a very natural solution to the
distortion problem; there is no distortion because there is no mapping. However,
existing techniques for synthesizing solid textures can be quite cumbersome. One

-2-

must learn how to "tweak" the parameters or procedures of the texture synthesizer to get a desired effect.

Therefore, a need exists in the art for a method and apparatus for generating textures that can cover, without seams or significant distortion, 5 surfaces of any size.

SUMMARY OF THE INVENTION

The present invention overcomes the disadvantages heretofore associated with the prior art by synthesizing image textures using an apparatus and a 10 concomitant method that synthesizes a texture of any size and shape from a given texture chip. Specifically, the apparatus generates a histogram of an input image (a texture chip) that represents texture parameters defining the texture properties of the input image, generates a histogram of a noise image having any size and shape, matches the noise image histogram to the input image histogram, 15 and modifies the noise image such that the modified noise image has a histogram that is the same as the histogram of the input image. The resultant modified noise image contains a texture that resembles the texture of the input image.

Further refinement of the texture in the modified noise image is accomplished by generating an image pyramid for the input image and an image 20 pyramid for the modified noise image. Thereafter, a histogram is generated for each subband in each of the pyramids and the histogram of the coarsest subband in the image pyramid of the modified noise image is matched to the histogram of the coarsest subband in the image pyramid of the input image. Once matched, the apparatus iterates, i.e., coarsest to finest, to the next subband in each image 25 pyramid and matches the histograms in each of the subbands. Ultimately, the apparatus modifies each image pyramid subband of the modified noise image such that each subband now has a histogram substantially identical to the histogram of the equivalent subband in the image pyramid of the input image. The pyramid for the modified noise image is then collapsed to generate a newly modified 30 (improved) noise image that better reflects the texture of the input image. The output image is a texture image having the texture properties of the input image and a size and shape defined by the size and shape of the noise image.

BRIEF DESCRIPTION OF THE DRAWINGS

35 The teachings of the present invention can be readily understood by considering the following detailed description in conjunction with the accompanying drawings, in which:

-3-

FIG. 1 is a high-level block diagram of a computer system used to execute the software routine(s) that implement the present invention;

FIG. 2 is a flow chart of an input image analysis routine and a texture synthesis routine used to implement the invention;

5   FIG. 3 is a functional block diagram of a transform used to generate a steerable image pyramid; and

FIG. 4 is a flow chart of a routine that modifies the noise image in response to texture parameters derived from the input image to generate a textured output image.

10   To facilitate understanding, identical reference numerals have been used, where possible, to designate identical elements that are common to the figures.

DETAILED DESCRIPTION

The present invention is apparatus and a concomitant method for
15   synthesizing an image (or solid texture) that matches the appearance of a given texture sample, e.g., a texture chip. One advantage of this invention is that it generates a texture for any size surface entirely from the texture sample, requiring no additional information or adjustment. The invention begins with a digitized image (e.g., the texture chip) and analyzes this input image to compute a
20   number of texture parameter values. Those parameter values are then used to synthesize a new (output) image of any size that looks (in its color and texture properties) like the input image. The analysis phase is inherently two-dimensional and, as such, the output image is typically two-dimensional. Nonetheless, in a three-dimensional application of the invention, the output image is a solid texture
25   such that planar slices through the solid have a texture like the original input image. In either case, the (two-dimensional or three-dimensional) texture is synthesized so that it tiles seamlessly.

FIG. 1 depicts a computer system 100 which executes software routines that form an illustrative implementation of the invention. Execution of the
30   software routines configure the computer system as an apparatus for generating image textures from an input image representing a sample of a texture that is to be generated within an output image of any shape and size. Specifically, the system 100 comprises a computer 102 and one or more input and output devices 104 and 106 for the computer. The computer may be a general purpose
35   computer having a central processing unit (CPU) 108 that is programmed to function in a certain manner by executing one or more programs residing in memory 110 (e.g., random access memory, floppy disk, hard disk and the like). The CPU functions in conjunction with well-known support circuits 112 such as

-4-

read only memory, power supplies, co-processors, and the like. The computer
system is controlled by conventional input devices, such as a mouse, keyboard,
trackball and the like, and displays information via conventional output devices,
such a computer graphics display, printer, plotter and the like. Although the input
5   image (e.g., the texture chip) may be stored in memory and recalled for use by the
system, typically, the input devices include a conventional image scanner that
converts a photograph, a frame of a video image, or half-tone image (or portion
thereof) into a digital image (input image 114). The display devices, on the other
hand, display the output image 116 as a stand-alone image or as a texture applied
10   to a surface in a graphical image.

In response to user commands applied to the computer via the input
devices, the CPU executes an input image analysis routine 118 that produces the
texture parameters and a texture synthesis routine 120 that generates the output
image 116 from the texture parameters. The output image may be displayed or
15   incorporated into another image using a conventional image rendering software
routine.

A. Texture Models

Textures have often been classified into two categories, deterministic
20   textures and stochastic textures. A deterministic texture is characterized by a set
of primitives and a placement rule (e.g., a tile floor). A stochastic texture, on the
other hand, does not have easily identifiable primitives (e.g., granite, bark, sand).
Many real-world textures have some mixture of these two characteristics (e.g.
woven fabric, woodgrain, plowed fields).

25   Much of the previous work on texture analysis and synthesis can be
classified according to what type of texture model was used. Some of the
successful texture models include reaction-diffusion, frequency domain, fractal,
and statistical/random field models. Some have used hybrid models that include a
deterministic (or periodic) component and a stochastic component. In spite of all
30   this work, scanned images and hand-drawn textures are still the principle source of
texture maps in computer graphics.

This invention focuses on the synthesis of stochastic textures. The
approach is motivated by research on human texture perception. Current
theories of texture discrimination are based on the fact that two textures are often
35   difficult to discriminate when they produce a similar distribution of responses in a
bank of (orientation and spatial-frequency selective) linear filters. The invention
described here, therefore, synthesizes textures by matching distributions (or
histograms) of filter outputs. This approach depends on the principle that all of

-4-

read only memory, power supplies, co-processors, and the like. The computer system is controlled by conventional input devices, such as a mouse, keyboard, trackball and the like, and displays information via conventional output devices, such a computer graphics display, printer, plotter and the like. Although the input
5   image (e.g., the texture chip) may be stored in memory and recalled for use by the system, typically, the input devices include a conventional image scanner that converts a photograph, a frame of a video image, or half-tone image (or portion thereof) into a digital image (input image 114). The display devices, on the other hand, display the output image 116 as a stand-alone image or as a texture applied
10   to a surface in a graphical image.

In response to user commands applied to the computer via the input devices, the CPU executes an input image analysis routine 118 that produces the texture parameters and a texture synthesis routine 120 that generates the output image 116 from the texture parameters. The output image may be displayed or
15   incorporated into another image using a conventional image rendering software routine.

A. Texture Models

Textures have often been classified into two categories, deterministic
20   textures and stochastic textures. A deterministic texture is characterized by a set of primitives and a placement rule (e.g., a tile floor). A stochastic texture, on the other hand, does not have easily identifiable primitives (e.g., granite, bark, sand). Many real-world textures have some mixture of these two characteristics (e.g. woven fabric, woodgrain, plowed fields).
25   Much of the previous work on texture analysis and synthesis can be classified according to what type of texture model was used. Some of the successful texture models include reaction-diffusion, frequency domain, fractal, and statistical/random field models. Some have used hybrid models that include a deterministic (or periodic) component and a stochastic component. In spite of all
30   this work, scanned images and hand-drawn textures are still the principle source of texture maps in computer graphics.

This invention focuses on the synthesis of stochastic textures. The approach is motivated by research on human texture perception. Current theories of texture discrimination are based on the fact that two textures are often
35   difficult to discriminate when they produce a similar distribution of responses in a bank of (orientation and spatial-frequency selective) linear filters. The invention described here, therefore, synthesizes textures by matching distributions (or histograms) of filter outputs. This approach depends on the principle that all of

-5-

the spatial information characterizing a texture image can be captured in the first
order statistics of an appropriately chosen set of linear filter outputs.
Computational efficiency is one of the advantages of this approach compared with
many of the previous texture analysis/synthesis systems. The inventive method
5   involves a sequence of image processing operations: convolution, subsampling,
upsampling, histogramming, and nonlinear transformations using small lookup
tables. These operations are fast, relatively simple to implement, and amenable
to special purpose hardware implementations (e.g., using digital signal processor
(DSP) integrated circuits) in lieu of a general purpose computer system.

10

B. Pyramid Texture Matching

FIG. 2 depicts a high-level flow chart of both the input image analysis
routine 118 and the texture synthesis routine 120 input image analysis routine
starts, at step 200, with an input (digitized) texture image (e.g., a texture chip).
15   The digitized image contains an array of pixels that appear as a texture such as
wood grain, sand, gravel, bricks, and the like. The texture chip is analyzed, at
step 202, to determine, at step 204, a plurality of texture parameters that
represent the texture contained in the texture chip.

The texture synthesis routine 120 begins, at step 206, with a noise image
20   (typically uniform white noise). The noise image is an array of pixels that have
randomly generated magnitudes. The size and shape of noise image is not
constrained by the invention and is typically user defined to fit a specific graphical
surface size and shape. Using the texture parameters from the input image
analysis routine, the texture synthesis routine modifies, at step 208, the noise
25   image to make it appear to have the texture of the texture chip. The resulting
synthetic image may have any amount of texturing. As shall be discussed below,
the foregoing processes are accomplished by making use of an invertible image
representation known as an image pyramid, along with an image modification
function that matches the histograms of two images. The following detailed
30   discussion of the invention presents examples using two types of pyramids: the
Laplacian pyramid (a radially symmetric transform) and the steerable pyramid
(an oriented transform). Before discussing the details of the invention, a review of
image pyramid construction and collapse is presented.

35   C. Image Pyramids

A linear image transform represents an image as a weighted sum of basis
functions. That is, the image, $I(x,y)$, is represented as a sum over an indexed
collection of functions, $g_i(x,y)$:

-6-

$$I(x, y) = \sum_i y_i g_i(x, y)$$

where y_i are the transform coefficients. These coefficients are computed from the signal by projecting onto a set of projection functions, $h_i(x, y)$:

$$y_i = \sum_{x,y} h_i(x, y) I(x, y)$$

For example, the basis functions of the Fourier transform are sinusoids and cosinusoids of various spatial frequencies. The projection functions of the Fourier transform are also (co-)sinusoids.

In many image processing applications, an image is decomposed into a set of subbands, and the information within each subband is processed more or less independently of that in the other subbands. The subbands are computed by convolving the image with a bank of linear filters. Each of the projection functions is a translated (or shifted) copy of one of the convolution kernels (see E.P. Simoncelli et al., SUBBAND IMAGE CODING, J.W. Woods, Ed., Klumer Academic Publishers, Norwell, MA (1990), for an introduction to subband transforms and image pyramids).

An image pyramid is a particular type of subband transform. The defining characteristic of an image pyramid is that the basis/projection functions are translated and dilated copies of one another (e.g., translated and dilated by a factor of 2 for some integer j). Although image pyramids are typically constructed through use of a factor of 2, any factor may be used. The subbands are computed by convolving and subsampling. For each successive value of j, in the typical implementation, the subsampling factor is increased by a factor of 2. This yields a set of subband images of different, progressive sizes (hence the name image pyramid) that correspond to different frequency bands.

In an independent context, mathematicians developed a form of continuous function representation called wavelets (see G. Strang, Wavelets and Dilation Equations: A Brief Introduction, SIAM 31 (1989), pp. 614-627, for an introduction to wavelets), that are very closely related to image pyramids. Both wavelets and pyramids can be implemented in an efficient recursive manner, as described below, and used interchangeably to implement the present invention.

Laplacian Pyramid. The Laplacian pyramid is computed using two basic operations: reduce and expand. The reduce operation applies a low-pass filter and then subsamples by a factor of two in each dimension. The expand operation upsamples by a factor of two (padding with zeros in between pixels) and then

-7-

applies the same low pass filter. A commonly used low-pass filter kernel (applied separably to the rows and columns of image) is: 1/16(1, 4, 6, 4, 1).

One complete level of the pyramid consists of two images, l_0 (a low-pass image), and b_0 (a high-pass image), that are computed as follows:

$$l_0 = \text{Reduce (im)}$$
$$b_0 = \text{im - Expand } (l_0),$$

where im is the original input image. Note that the original image can be trivially reconstructed from l_0 and b_0:

$$\text{reconstructed-im} = b_0 + \text{Expand}(l_0).$$

The next level of the pyramid is constructed by applying the same set of operations to the l_0 image, yielding two new images, l_1 and b_1. The full pyramid is constructed (via a "make-pyramid" function) by successively splitting the low-pass image l_i into two new images, l_{i+1} (a new low-pass image) and b_{i+1} (a new band-pass image).

The combined effect of the recursive low-pass filtering and sub/upsampling operations yields a subband transform whose basis functions are (approximately) Gaussian functions. In other words, the transform represents an image as a sum of shifted, scaled, and dilated (approximately) Gaussian functions. The projection functions of this transform are (approximately) Laplacian-of-Gaussian (mexican-hat) functions, hence the name Laplacian pyramid. Note that the pyramid is not computed by convolving the image directly with the projection functions. The recursive application of the reduce and expand operations yields the same result, but much more efficiently.

The result is a collection of pyramid subband images consisting of several bandpass images and one leftover lowpass image. These images have different sizes because of the subsampling operations; the smaller images correspond to the lower spatial frequency bands (coarser scales). Note that the original image can always be recovered from the pyramid representation (via a "collapse-pyramid" function) by inverting the sequence of operations, as exemplified above.

Steerable Pyramid. Textures that have oriented or elongated structures are not captured by the Laplacian pyramid generation process because its basis functions are (approximately) radially symmetric.

To synthesize anisotropic textures, the invention uses a steerable pyramid transform. Like the Laplacian pyramid, this transform decomposes the image

into several spatial frequency bands. In addition, it further divides each frequency band into a set of orientation bands.

FIG. 3 shows a functional block diagram of apparatus for computing the steerable pyramid transform. The left-hand side 300 of the figure is an analysis portion (make-pyramid) and the right hand side 304 is a synthesis portion (collapse-pyramid). The circles 302 in between the two portions 300 and 304 represent the decomposed subband images. The transform begins with a high-pass/low-pass split 308 using a low-pass filter with a radially symmetric frequency response and a high-pass filter 312 having a response that corresponds to the four corners of the spatial frequency domain. Each successive level of the pyramid is constructed from the previous level's low-pass band by applying a bank of band-pass filters 314 and a low-pass filter 316.

The orientation decomposition at each level of the pyramid is "steerable", that is, the response of a filter tuned to any orientation can be obtained through a linear combination of the responses of the four basis filters computed at the same location. The steerability property is important because it implies that the pyramid representation is locally rotation-invariant.

The steerable pyramid, unlike most discrete wavelet transforms used in image compression algorithms, is non-orthogonal and over-complete; the number of pixels in the pyramid is much greater than the number of pixels in the input image (note that only the low-pass band is subsampled). This is accomplished to minimize the amount of aliasing within each subband. Avoiding aliasing is critical because this pyramid-based texture analysis/synthesis process treats each subband independently.

The steerable pyramid is self-inverting; the filters in the synthesis portion 304 of the apparatus are the same as those in the analysis portion 300 of the apparatus. This allows the reconstruction (synthesis side) to be efficiently computed despite the non-orthogonality.

Although the steerable pyramid filter kernels are nonseparable, any nonseparable filter can be approximated (often quite well) by a sum of several separable filter kernels. Using these separable filter approximations would further increase the computational efficiency. Software implementations of steerable pyramid generation apparatus are readily available in the art.

Psychophysical and physiological experiments suggest that image information is represented in visual cortex by orientation and spatial-frequency selective filters. The steerable pyramid captures some of the oriented structure of images similar to the way this information is represented in the human visual system. Thus, textures synthesized with the steerable pyramid generally appear

-9-

noticeably better than those synthesized with the Laplacian pyramid or some other non-oriented representation. Other than the choice of pyramid type, the processes used by the invention are exactly the same.

5   Returning to FIG. 2, to facilitate input image analysis and texture synthesis the input image as well as the modified noise image are converted into image pyramid representations. The image pyramid conversion of the input image is typically accomplished within the input step (step 200) while conversion of the modified noise image occurs within step 208. The choice of a steerable pyramid or a Laplacian pyramid or, for that matter, any other pyramid or wavelet

10   representation of the images, depends on the texture being synthesized. For example, a texture having elongated or oriented features may require a steerable pyramid to best represent that image. The specific implementation software used to generate image pyramids and/or wavelet representations of images is well known in the art.

15

D. Histogram Matching Function

Modification of the noise pyramid is accomplished using a histogram matching function. Histogram matching is a generalization of histogram equalization. The invention takes an image (noise image) and coerces it via a pair

20   of lookup tables to have a particular histogram, e.g., the noise image is modified. The two lookup tables are: (a) the cumulative distribution function (cdf) of one image (noise image), and (b) the inverse cumulative distribution function (inverse cdf) of the other image (texture chip).

An image's histogram is computed by choosing a bin-size (typically, but not

25   necessarily, using 256 bins), counting the number of pixels having magnitudes that fall into each bin, and dividing by the total number of pixels in the image. An image's cdf is computed from its histogram simply by accumulating successive bin counts. As such, a histogram represents the texture of the input image, i.e., the histogram is, in essence, an array of texture parameters.

30   The cdf is a lookup table that maps from the interval [0,256] to the interval [0,1]. The inverse cdf is a lookup table that maps back from [0,1] to [0,256]. It is constructed by resampling (with linear interpolation) the cdf so that its samples are evenly spaced on the [0,1] interval. Of course, these particular intervals are illustrative and the invention can be used to map between any two arbitrarily

35   selected intervals.

These two lookup tables are used by the match-histogram function to modify an image (im1) to have the same histogram as another image (im2). The match histogram process is repeated until a predefined criterion is met such as

the achieving an identity look up table. The following pseudocode illustrates the
match histogram function.

```
Match-histogram (im1, im2)
        im1-cdf = Make-cdf (im1)
        im2-cdf = Make-cdf (im2)
        inv-im2-cdf= Make-inverse-lookup-table (im2-cdf)
        Loop for each pixel do
                im1[pixel] = Lookup (inv-1m2-cdf, Lookup (im1-cdf, im1[pixel]))
                        PSEUDO CODE 1
                     Match Histogram Function
```

E. Texture Synthesis

        FIG. 4 depicts a detailed flow chart of the routine that implements a
"match-texture" function to iterively modify an input noise image so that it looks
like an input texture image. To accomplish this function, the invention, at
step 404, matches the histogram of the noise image (produced at step 402) to the
histogram of the input image (provided at step 400 from the input image analysis
routine). At step 406, the invention modifies, using PSUEDOCODE 1 above, the
noise image to conform it's histogram to the histogram of the input image.

        At step 408, the routine generates an image pyramid representation of the
modified noise image. Additionally, the image pyramid of the input image is
provided at step 410. The image pyramids may be steerable or Laplacian
depending upon the content of the texture, i.e., if the texture has elongated or
oriented features, a steerable pyramid may be more appropriate.

        Respectively at steps 412 and 414, the routine generates a histogram for
each level (subband) of each respective pyramid. The routine loops through the
two pyramid data structures and matches the histograms of each of the
corresponding pyramid subbands. After each matching step (step 416), the
routine modifies, at step 418, the subband of the modified noise image to conform
its histogram to that of the same subband in the input image. Both the matching
and modifying steps are represented in PSEUDOCODE 1 above, where each
subband is treated as an image, im1 and im2. At step 420, the routine queries
whether histogram of the last subband in the image pyramid of the noise image is
being matched to the input image subband. If the query is negatively answered,
the routine increments the subband at step 422 and returns to step 416 to
process the next subband, e.g., match histogram in the newly selected subband. If
the query is affirmatively answered the routine proceeds to step 424 to collapse

-11-

the pyramid of the modified noise image to form an updated modified noise image
that generally represents the texture. At step 426, the routine queries whether a
stop criterion has been met such as a look up table approaching or attaining
identity values. If this query is negatively answered, the routine returns to step

5   404, where the modified noise image generated in step 424 is processed in the
same manner that the noise image was processed the first time through the loop.
In this manner, the routine iteratively updates (corrects) the modified noise image
until the stop criterion indicates that the synthesized image contains a texture
that is substantially similar to that of the input image. Upon meeting the stop

10   criterion at step 426, the routine returns, at step 430, to the texture synthesis
routine of FIG. 2. The foregoing process is illustrated in the following pseudocode,
where an a-band is a subband of the analysis pyramid (input image) and the s-
band is a subband of the synthesis pyramid (modified noise image).

15   Match-texture (noise, texture)
              Match-Histogram (noise, texture)
              analysis-pyr = Make-Pyramid (texture)
              Loop for several iterations do
                      synthesis-pyr = Make-Pyramid (noise)
20   Loop for a-band in subbands of analysis-pyr and for
                              s-band in subbands of synthesis-pyr do
                              Match-Histogram (s-band, a-band)
                      noise = Collapse-Pyramid (synthesis-pyr)
              Match-histogram (noise, texture)
25   PSEUDO CODE 2
                              Match Texture Function

          Whenever an iterative scheme of this sort is used there is a concern about
convergence. Therefore, stopping execution of the routine after several (five or so)
30   iterations is generally necessary. As is the case with nearly all discrete filters,
there are tradeoffs in the design of the steerable pyramid filters (e.g., filter size
versus reconstruction accuracy). Since the filters are not perfect, iterating too
many times introduces artifacts due to reconstruction error.
          The core of the invention is histogram matching which is a spatially local
35   operation. The primary reason that this spatially local operation reproduces the
spatial characteristics of textures is that histogram matching is accomplished on
a representation that has intrinsic spatial structure. A local modification of a
value in one of the pyramid subbands produces a spatially correlated change in the

-12-

reconstructed image. In other words, matching the pointwise statistics of the pyramid representation does match some of the spatial statistics of the reconstructed image. Clearly, only spatial relationships that are represented by the pyramid basis functions can be captured in this way so the choice of basis

5   functions is important. As mentioned above, the steerable pyramid basis functions are a reasonably good model of the human visual system's image representation and, as such, these basis functions are generally preferred.

Using a complete model of human texture perception could presumably lead to perfect texture matches. By analogy, the present understanding of the

10 wavelength encoding of light in the retina allows matching the color appearance of (nearly) any color image with only three colored lights (e.g., using an RGB monitor). Lights can be distinguished only if their spectral compositions differ in such a way as to produce distinct responses in the three photoreceptor classes. Likewise, textures can be distinguished only if their spatial structures differ in such a way as

15 to produce distinct responses in the human visual system.

F. Edge Handling

Proper edge handling in the convolution operations is also important to accurately synthesizing a texture that will be tiled. For the modified noise image

20 pyramid, the invention uses circular convolution. In other words, for an image $(I(x,y))$ of size NxN, define: $I(x,y) = I(x \bmod N, y \bmod N)$. Given that the synthesis starts with a random noise image, circular convolution guarantees that the resulting synthetic texture will tile seamlessly.

For the input image pyramid, on the other hand, circular convolution would

25 typically result in spuriously large filter responses at the image borders. This would, in turn, introduce artifacts in the synthesized texture. A reasonable border handler for the input image pyramid is to pad the image with a reflected copy of itself. Reflecting at the border usually avoids spurious responses (except for obliquely oriented textures).

30

G. Alternatives

One can imagine a number of other ways to mix/combine texture: to synthesize an image that looks a bit like each of the input textures: apply the match-texture function to a second image that already has a texture rather than

35 noise, combine the high frequencies of one texture with the low frequencies of another, combine two or more textures by averaging their pyramid histograms, and the like.

There is no general way to construct an anisotropic solid texture from a two-dimensional sample. However, there are several options including: (1) constructing a solid texture as the outer product of a two-dimensional anisotropic color texture image and a 1-d (monochrome) signal; (2) composing (adding,

5 multiplying, and the like) several solid textures; (3) starting with an isotropic solid, and introducing anisotropy procedurally; (4) starting with an isotropic solid, and using a paint program to introduce anisotropic "touch-ups".

Image pyramids and multi-scale image representations of one sort or another are the most often used data structures for antialiased texture mapping.

10 Pyramid-based texture synthesis, therefore, can be naturally integrated into an antialiased texture mapping system.

The invention can be incorporated into an interactive tool for texture synthesis, with a "slider" for each parameter in the pyramid representation. In the current implementation, each subband histogram is encoded with 256 bins.

15 However the subband histograms of many "natural" images have a characteristic shape, suggesting that a very small number of parameters may be sufficient and easily controlled by a tool.

The foregoing embodiment was discussed as generating image pyramids without regard for image color, i.e., processing monochrome images. Those skilled

20 in the art will understand that the invention can be used to process color images by separating each of the images into their constituent color images and producing three pyramids per image, e.g., one pyramid each for red, green and blue (RGB) colors or cyan, magenta, yellow (CMY) colors. All processing of the invention is accomplished on each image pyramid such that three image pyramids of the

25 modified noise image are ultimately produced as output images. These three pyramids are then collapsed and combined to form a final color image as a color texture output image.

Although one embodiment incorporating the teachings of the present invention has been shown and described in detail herein, those skilled in the art can

30 readily devise many other varied embodiments that still incorporate these teachings.

What is claimed is:

1   1. A method for generating image textures comprising the steps of:
2   (a) providing an input image containing an array of pixels representing a
3   texture;
4   (b) providing a noise image containing an array of pixels representing noise;
5   (c) analyzing the input image to determine texture parameters representing
6   the texture in said input image;
7   (d) modifying, in response to the texture parameters, the noise image to
8   synthesize a modified noise image, wherein said noise image attains the texture
9   parameters of the texture in said input image; and
10   (e) generating said modified noise image as an output image.
11

1   2. The method of claim 1 wherein said analyzing step further comprises the step
2   of generating a histogram of the input image, where said histogram contains said
3   texture parameters.
4

1   3. The method of claim 2 wherein said analyzing step further comprises the steps
2   of:
3   generating a histogram of the noise image;
4   matching the noise image histogram to the input image histogram; and
5   modifying the noise image to form a modified noise image, where said
6   modified noise image attains the histogram of the input image.
7

1   4. The method of claim 1 further comprising the steps of:
2   (f) generating an image pyramid of said modified noise image containing a
3   plurality of first subbands;
4   (g) generating an image pyramid of said input image containing a plurality
5   of second subbands;
6   (h) generating a histogram for each of said subbands in said plurality of first
7   subbands;
8   (i) generating a histogram for each of said subbands in said plurality of
9   second subbands;
10   (j) selecting a subband from each of said plurality of first and second
11 subbands;
12   (k) matching said histogram of said selected first subband with said
13 histogram of said selected second subband;

14   (l) modifying said selected subband of said image pyramid of said modified
15 noise image to attain said histogram of said selected subband of said image
16 pyramid of said input image; and
17   (m) collapsing said modified image pyramid having a modified subband to
18 form a second modified noise image.
19

1 5. The method of claim 4 wherein said (j) selecting step is repeated in a coarse to
2 fine subband order and wherein said (k) matching and (l) modifying steps are
3 repeated for each selected subband.
4

1 6. The method of claim 5 further comprising the step of:
2   (n) repeating steps (f), (h), (j), (k), (l), and (m) using said second modified
3 image as said modified noise image until a predefined criterion is met.
4

1 7. Apparatus for generating image textures having as inputs an input image
2 containing an array of pixels representing a texture and a noise image containing
3 an array of pixels representing noise, said apparatus comprising:
4   means for analyzing the input image to determine texture parameters
5 representing the texture in said input image;
6   means for modifying, in response to the texture parameters, the noise
7 image to synthesize a modified noise image, where said noise image attains the
8 texture parameters of the texture in said input image; and
9   means for generating said modified noise image as an output image.
10

1 8. The apparatus of claim 7 wherein said means for analyzing further comprises a
2 means for generating a histogram of the input image, where said histogram
3 contains said texture parameters.
4

1 9. The apparatus of claim 8 wherein said means for modifying further comprises:
2   means for generating a histogram of the noise image;
3   means for matching the noise image histogram to the input image
4 histogram; and
5   means for modifying the noise image to form a modified noise image, where
6 said modified noise image attains the histogram of the input image.
7

1 10. The apparatus of claim 7 further comprising:
2   noise pyramid means for generating an image pyramid of said modified noise
3 image containing a plurality of first subbands;

4   input pyramid means for generating an image pyramid of said input image
5   containing a plurality of second subbands;
6   first subband means, connected to said noise pyramid means, for generating
7   a histogram for each of said subbands in said plurality of first subbands;
8   second subband means, connected to said input pyramid means, for
9   generating a histogram for each of said subbands in said plurality of second
10   subbands;
11   means, connected to said first and second subband means, for selecting a
12   subband from each of said plurality of first and second subbands;
13   means, connected to said means for selecting, for matching said histogram
14   of said selected first subband with said histogram of said selected second subband;
15   means, connected to said means for matching, for modifying said selected
16   subband of said image pyramid of said modified noise image to attain said
17   histogram of said selected subband of said image pyramid of said input image;
18   means, connected to said means for modifying, for collapsing said modified
19   image pyramid having a modified subband to form a second modified noise image.
20

1   11. The apparatus of claim 10 further comprising means for repetitively
2   operating said means for selecting to select subbands in a coarse to fine subband
3   order and, for each selected subband, repetitively operating said means for
4   matching and said means for modifying.
5

1   12. The apparatus of claim 10 further comprising means for repetitively
2   operating said noise pyramid means, first subband means, means for selecting,
3   means for matching, means for modifying, and means for collapsing using said
4   second modified image as said modified noise image until a predefined criterion is
5   met.
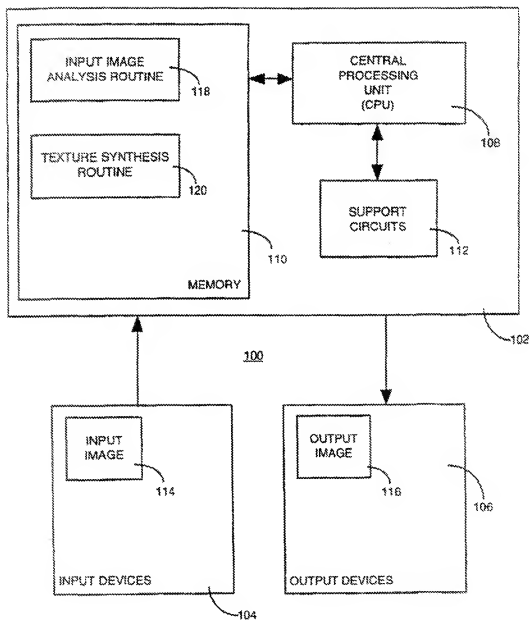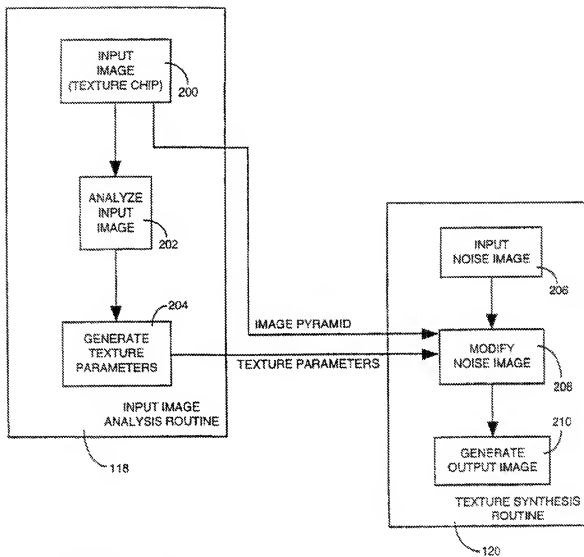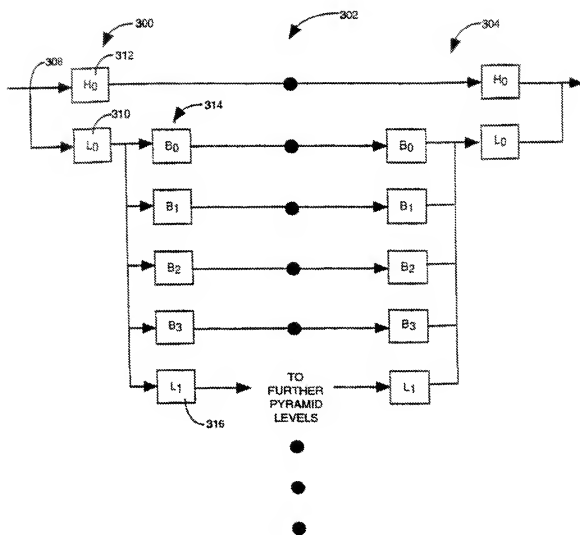6

FIG. 1

FIG. 2

FIG. 3

FIG. 4